REMARKS

Reconsideration and allowance of the above-referenced application are respectfully requested. Upon entry of this amendment, claims 1-73 will be pending.

Applicant thanks the Examiner for the indication that claims 22-28 and 30-38 contain allowable subject matter.

The Examiner objects to the abstract of the disclosure because it contains more than 150 words. Correction has been made.

Claims 42-45 are rejected under 35 USC 112, second paragraph, as being indefinite. Claim 41 recites the limitation "compression engine," and applicant's attorney, making a typographical error, inserted the phrase ""search engine" into claims 42 and 43. Correction of this error has been made to claims 42 and 43.

Claims 1-3, 6-7, 16-18, 20-21, 29 and 39-46 are rejected under 35 USC 102(e) as being anticipated over Johns 6,366,289. Further claims 4-5, 11-13 And 19 are rejected under 35 USC § 103(a) as being unpatentable over Johns 6,366,289 in view of Simms 5,586,280. and claims 8-10 and 14-15 are rejected under 35 USC § 103(a) as being unpatentable over Johns 6,366,289 in view of Morikawa et al. 6,043,897. Applicant respectfully traverses the Examiner's grounds of rejection.

Claim 1 and claims dependent thereon

Independent claim 1 is directed to a method of operating upon digital data to obtain compression in a specifically defined manner, in particular, partitioning the digital data into a plurality of blocks, creating a plurality of first threads, such that each first thread includes at least one of the plurality of blocks, and operating upon each of the plurality of first threads to obtain a plurality of compressed first threads. (Emphasis Added)

Accordingly, the step of "creating a plurality of first threads" specifically recites the creation of independent threads that a processor or processors that support threads can operate upon independently. As further recited in the claim, associated with each of these first threads is at least one of the plurality of blocks of digital data. Therefore, the step of "operating upon each of the first threads" requires execution of operations associated with each first thread, with the result being that once operations are completed on a particular thread, there exists compressed blocks of digital data associated with that completed thread (recited as one of the compressed first threads since the completed thread has compressed digital data associated with it).

The Examiner, instead of reading the term "thread" as a processor operation, which was the context clearly recited, instead uses the term thread to refer to a portion of data –and then associated that reading with the concept of "chunks" as recited in Johns.

Applicant respectfully submits that this interpretation of "threads" is incorrect, for a number of reasons.

First, these claims, and the specification, clearly refer to threads being associated with processor operation, which in turn operates upon the data associated with the processor operation on that thread. When initially introducing the compression/decompression engine 230 in the detailed description of the preferred embodiments in the specification at page 7, lines 4-6, it is clearly recited that the compression/decompression engine is implemented "using at least one processor that has the capability of operating upon multiple threads simultaneously." (Emphasis Added) And "threads" being operated upon by a processor in this context is used consistently thereafter. In particular, at page 13, the paragraph beginning at line 7 specifically describes "the concept of using different threads for the compression of different data" according to the invention. In the following description, it is taught:

> Accordingly, if there is a certain block that interface controller 220 has predicted will be difficult to compress, then a separate *thread* can be identified, with that *thread* having associated with it unique metadata as well as control signals, that provide the information necessary for the C/D engine 230 to begin the compression routine operation on that thread. Accordingly, for each block that interface controller 220 has determined should be independently compressed, a separate *thread* will be created.
>
> Since the present invention can operate independently on each block if needed, although in many instances it will operate on many blocks with a single *thread*, the interface controller needs to be able to determine when to create a new thread or when to use the same *thread* for multiple blocks, as shown by step 360 in Fig. 3. For instance, if the time needed to compress a block of data is greater than some threshold value, then a new *thread* is created for that block is created by the interface controller 220. If not, then another block is added to the previous block, such that a string of blocks will be tagged for being compressed via the same *thread* by the interface controller 220. (Emphasis Added)

The above usage in the specification also conforms with the commonly understood meaning of a "thread," which requires the ability of a processor to execute multiple different sequences of instructions. For example, "thread" is defined on webopedia.com as "[I]n programming, a part of a program that can execute independently of other parts. Operating systems that support

multithreading enable programmers to design programs whose threaded parts can execute concurrently."

Clearly, as used in the specification, and as would have been interpreted by one of ordinary skill in the art, a "thread" is clearly distinct from "blocks" of data or "strings" of data, and relates to processor execution. This construction is also consistent with the manner in which claims are intended to be interpreted by the Patent Office. In fact, the MPEP quotes *In re Morris* for the proposition that when interpreting claims during examination, the "PTO applies to verbiage of the proposed claims the broadest reasonable meaning of the words in their ordinary usage as they would be understood by one of ordinary skill in the art, taking into account whatever enlightenment by way of definitions or otherwise that may be afforded by the written description contained in applicant's specification." *In re Morris*, 127 F.3d 1048, 1054-55, 44 USPQ 2d 1023, 1027-28 (Fed. Cir. 1997).

Upon re-reading the specification, while applicant found three instances (out of many instances of this term in the detailed description) in which the term thread was used instead of the phrase "string of blocks" or just "blocks," these three instances have been corrected, are clearly the exception to the manner in which this term has been used, and should not form the basis for an improper interpretation of the term "threads" as used in the claims. And the statements made herein further confirm the appropriate interpretation.

Once interpreted correctly, it is apparent that Johns clearly does not teach or suggest the concept of "threads." Johns nowhere uses the term "thread." Apparently conceding this lack of usage of the term, the Examiner clearly indicates in the Office Action that the recitation of "threads" is interpreted as "chunks of data." That chunks of data is inconsistent with the appropriate interpretation of a "thread" is apparent, since there is no relationship between processor operation and chunks of data. Accordingly, applicant respectfully submits that the improper hindsight is being used in an inappropriate attempt to show that the claims do not contain patentable subject matter.

Still further, even if the Examiner continues to contend that a broader interpretation of "threads" as being only a data portion is appropriate, Johns nonetheless still does not have "threads" according to this interpretation. That is the case because there is no unit of data smaller than a block recited in the claims. In Johns. however, "chunks" are a subset of the block. Thus, even using the Examiner's inappropriate interpretation, each thread would require one or multiple

blocks, the operation upon a "chunk," which is smaller than a block, does not meet the claim limitations as recited.

That improper hindsight is being used, as remarked above, is further evidenced by the rejections of the dependent claims.

For example, for the concept of threads being operated upon in parallel in claim 4, the Examiner does not even use Johns, but instead another reference --Simms. Whereas Johns is directed to a method and system for managing an electronic display image using compressed and uncompressed blocks, Simms is directed to a method and apparatus for appending data to compressed records previously stored on a sequentially-accessible storage medium.

How one of ordinary skill would be motivated to combine an invention directed to a method for managing compression and decompression of an electronic display image stored in a video memory with an invention that is essentially a printer without impermissibly using hindsight, is not apparent to the applicant. And even if combined, the present inventions would not result.

That hindsight is being used is even further shown by the manner in which Simms is being used to show "threads" allegedly being operated upon in parallel. The language from Simms referred to in the office action is:

"[F]or data which is being compressed, in parallel with organization of the compressed data into groups, the entity manager generates ... (col. 17, lines 16-19); and

"[I]n parallel with the generation of the main data blocks, 35-byte sub blocks are also generated that contain... (col. 19, lines 27-28).

Clearly, these disparate teachings at columns 17 and 19 do not suggest operating upon different "threads" in parallel. Their only commonality is the use of the word "parallel" in the passage. Thus, the language cited only recites that with one operation being performed (compression in col. 17 or generation of main data blocks in col. 19), another different operation (not having anything to do with compression or main data block generation, respectively) will also be performed. And while there is commonality in both passages using the word "parallel," there is no connotation of parallel being used in the sense of a processor capable of operating upon multiple threads simultaneously.

Claim 19 is also rejected (with respect to the second threads) using the same column and line references to Johns and Simms used in claim 4. While the fact that the second threads are not distinguished from the first threads is sufficient for patentability, (as well as for other reasons --see

comments hereinafter with respect to claim 17), the usage of second threads being operated upon in parallel is a further patentable distinction.

Another illustration of inappropriate hindsight is with respect to claim 17. Claim 17, dependent on claim 1, recites the additional steps of

operating upon each of the compressed first threads to eliminate each of the compressed first threads and retain the compressed first blocks;

creating a plurality of second threads, such that each second thread includes at least one of the plurality of compressed first blocks; and

operating upon each of the plurality of second threads to obtain a plurality of compressed second threads, each compressed second thread including at least one compressed second block of digital data.

In addition to clearly reciting additional threads in a manner as commented upon previously with respect to claim 1, this claim further recites that the created second threads include at least one of the plurality of compressed first blocks. Applicant respectfully asserts that the Examiner pointing to the identical passages of Johns that he did with respect to these second threads as he did for the first threads recited in claim 1 illustrates that Johns is an inappropriate reference. And applicant furthermore, respectfully asserts that that the Examiner pointing to the identical passages of Johns that he did with respect to with respect to operating upon the second threads (which according to even the examiner's logic must contain compressed blocks) to obtain second compressed blocks as he did for the step of operating upon the first threads recited in claim 1 illustrates that Johns is an inappropriate reference.

And, even furthermore, the contorted construction of "eliminating each of the first threads as recited" to mean "freeing up memory" as stated by the Examiner in his office action, very respectfully, makes no sense whatsoever. One step of the claim recites eliminating each first thread, but retaining the first compressed blocks associated with each first thread. The passage at column 2, line 66 to column 3, line 3 of Johns where "freeing up memory" is used merely states that the controller in Johns leaves data in an uncompressed format until it is necessary to free up memory, and that the system of Johns "compresses least recently used [decompressed] blocks when necessary to free up memory for other decompressed blocks." Since the claim requires retaining the compressed first blocks, and then performing subsequent steps using those compressed first blocks, the passage in Johns that teaches compressing decompressed data to "free up memory"

clearly has no applicability. Again, this illustrates the improper usage of hindsight.

Other claims dependent on claim 1 are also clearly not anticipated or rendered obvious by Johns, either alone or in combination with the other references. Claims in which only Johns is used as a reference will be first discussed.

For example, claim 7 recites that the step of creating each of the first threads associates blocks with threads so that each of the first threads shares certain common compression characteristics. Column 7, lines 62-66 of Johns merely refers to accessing compressed chunks using an address. And furthermore, there is no teaching or suggestion in Johns of data in various blocks having different compression characteristics –which is not surprising, since the entirety of Johns relates to compressing and decompressing only images.

Claim 16 recites that the step of partitioning data includes the step of determining the size of each of the plurality of blocks taking data type into account. Column 10, lines 45-59 referred to by the Examiner merely assert that the VFB controller can be implemented to select the size of the memory block based on the pixel format, or using a fixed block size. Given that a particular display will only be able to use a single type of pixel format (as acknowledged in 52-5 of that same column) clearly there is only a single data type that is being operated upon by Johns, and there is no capacity for selecting the size of various blocks to be different based upon blocks containing different data types.

Claim 18, dependent upon claim 17 previously discussed, recites that each of second threads are operated upon independently. Again, the Examiner points to the same passage in Johns that was used in rejecting claim 2, which deal with operating upon the first threads independently. That the two are distinct are apparent, but they have clearly not been treated as distinct.

Claim 20 recites that the same compression algorithm used to operate upon the first blocks is used to operate upon the compressed block. While Johns at column 17, lines 45-51 teaches various compression methods, there is clearly no teaching or suggestion of using the same compression method to operate upon both blocks and also compressed blocks. The comment in the office action that "[I]t is inherent that the same compression algorithm can be used to operate upon each block [of Johns]" also suggest that this claim, as well as potentially claims 17-38, have not been correctly understood by the Examiner.

Claim 29 recites that each first thread has an associated first metadata set. Johns column 6, lines 1-6 merely states that the amount of memory consumed by a chunk will depend upon a

number of factors, including compression format and pixel format. Nowhere in this passage is there any mention of metadata.

Other claims dependent on claim 1 were rejected based upon a combination of references.

The inappropriateness of the combination of Johns and Simms was commented upon previously with respect to claims 4 and 19. This is true of other claims rejected based upon this combination.

For example, claim 11 recites using the data type to create the first plurality of threads. That Johns has no need for this is apparent, since all the data is for electronic image display, and thus is of the same type. That the records being appended to in Simms are also of the same type is also apparent. Thus, clearly there is no need for this feature in either reference. Furthermore, the reference to column 7, lines 11-16 of Simms to the existence of a block access table neither teaches nor suggests this recited step, since there is not even a "data type" bit for any of the FLAG entry bits, and using the usage of data type in the context recited by this claim is clearly not contemplated.

Claim 12 refers to a data type in the header, and clearly the mention of "entity header portion" at column 3, lines 7-14 of Simms does not teach or suggest this concept The header information discussed therein only relates to the number of wanted records –all of which are clearly of the same data type.

Claim 13 recites comparing block data to various predetermined data patterns to determine data type, and the reference to Simms at column 20, lines 8-24 is a reference to the writing process, the generation of track signals, and the avoidance of flux transitions. Nothing in this passage remotely resembles comparing data blocks to predetermined data patterns.

As the combination of Johns and Simms is not appropriate, similarly, the combination of Johns and Morikawa is not appropriate.

Whereas Johns is directed to a method and system for managing an electronic display image using compressed and uncompressed blocks, Morikawa is directed to a method and apparatus for digital copying.

How one of ordinary skill would be motivated to combine an invention directed to a method for managing compression and decompression of an electronic display image stored in a video memory with an invention directed to appending new data to previously compressed records without impermissibly using hindsight is not apparent to the applicant. And even if combined, the present inventions would not result.

For example, claims 8 and 14 require predicting an estimated compression time and estimated compression amount for each block. Here the Examiner refers to Morikawa at column 2, lines 14-18 and column 5, lines 57-63. While these passages recite reference to compression times, expansion times, and the size of the compressed image, there is no teaching or suggestion for predicting an estimated compression time and estimated compression amount for each block.

And claims 9 and 15, dependent on claims 8 and 14, respectively, require using the estimated compression time and estimated compression amount to determine which blocks should be associated with the same first thread. In addition to Johns not properly being modifiable according to claims 8 and 14, clearly the introductory passage at column 2, lines 14-18 Morikawa adds nothing to teach or suggest using such predictions to determine which blocks should be associated with the same first thread.

Independent Claims 39 and 40, and claims dependent thereon

Independent method and apparatus claims 39 and 40 are directed to performing compression in multiple passes, and decompression in a single pass.

In rejecting these claims solely based upon Johns, column 5, lines 45-47 is used for the alleged teaching of compression in multiple passes, and column 5, lines 47-48 is used for the alleged teaching of decompression in a single pass. The entirety of the two sentences that is within these lines reads:

The invention provides a method for managing display memory where a visible display image as well as other images are partially compressed and partially uncompressed

Nothing is mentioned about compressing in a multiple passes to obtain compressed digital data. Nothing is mentioned about decompressing the compressed digital data in a single pass. Clearly, nothing is mentioned whatsoever that teaches or suggests the subject matter of these claims.

Claims dependent thereon also contain allowable subject matter. For example, that each of a plurality of central processing units operate upon a different plurality of threads as recited in claim 44 is not taught or suggested by Johns.

Claim 46

Claim 46 recites the step of obtaining metadata representative of patterns in first digital data from the compression of the first digital data in a first compression system, and then distributing the metadata to a second compression system for use by the second compression system in compressing second digital data.

The reference to the Abstract of Johns for the proposition that that Johns teaches a "method of allowing a plurality of compression systems to operate more efficiently" is simply not understood, and the reference to column 6, lines 1-6 and 29-35 are simply references to the fact that in different systems the amount of memory used will be different depending upon the compression format and the pixel format, as well as the relationship between a virtual frame buffer and physical address space. There is no teaching or suggestion of metadata at all, and clearly no teaching or suggestion to use metadata obtained from one compression system in another compression system. Accordingly, claim 46 contains allowable subject matter.

Applicant notes that new claims 47-73 have been added, and these newly added claims also contain patentably distinct subject matter.

In view of the above amendments and remarks, applicant submits that the above-referenced application is in a condition for allowance, and such a notice is respectfully requested.

Respectfully submitted,
PILLSBURY WINTHROP LLP

By: _____
David A. Jakopin, Reg. No. 32,995

1600 Tysons Boulevard
McLean, Virginia 22102
Tel. No.: (650) 233-4790
Fax No.: (650) 233-4545

60325401